

2장 서울시 범죄 현황 분석(110-121쪽)

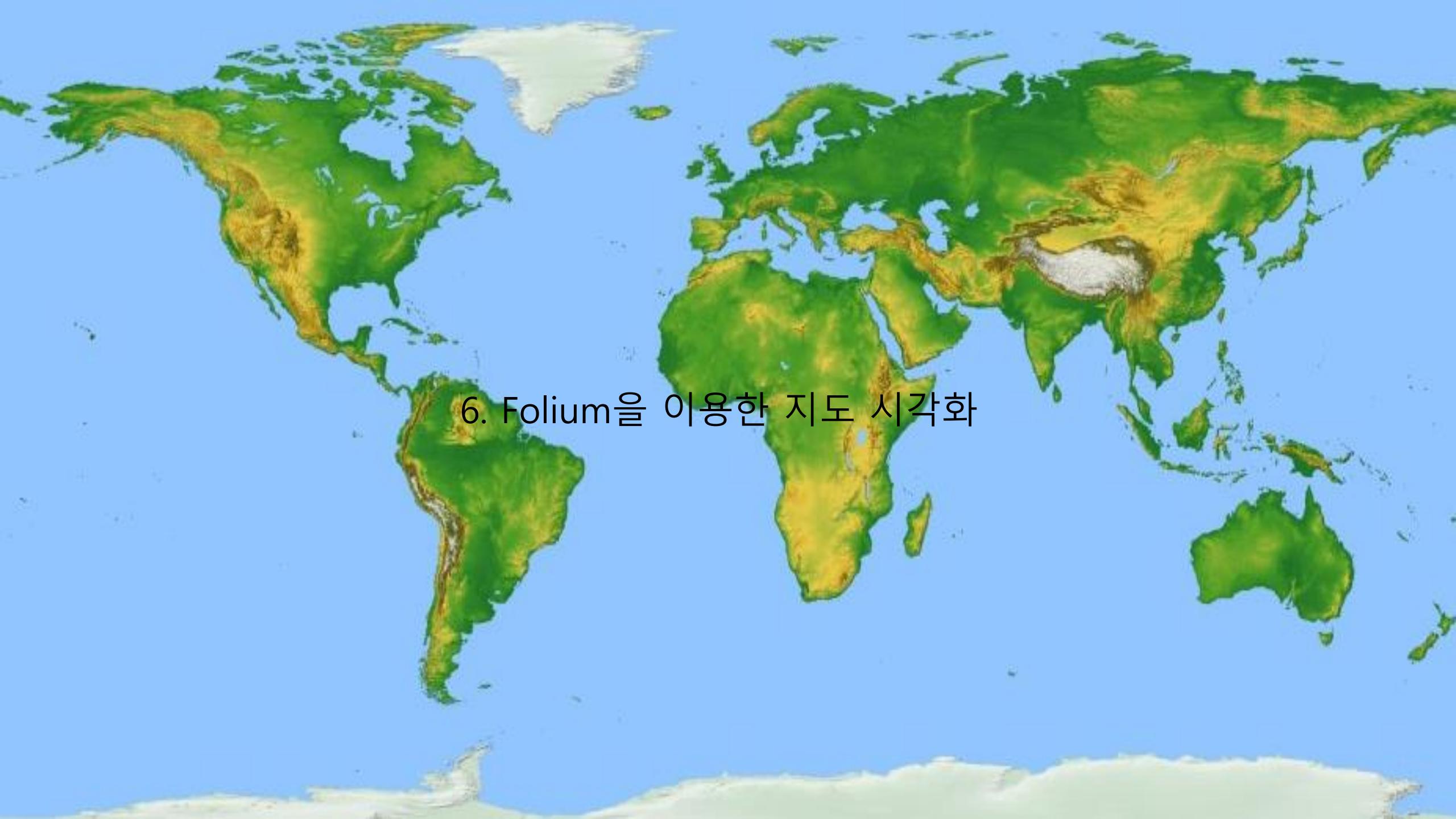
1. Pandas를 이용하여 데이터 정리하기
2. Google Maps 이용하여 지도 활용하기
3. Pivot_table을 이용한 데이터 정리
4. Seaborn을 이용한 시각화
5. 범죄데이터 시각화

2019.09.24~2019.09.25 과제

-
-
6. Folium을 이용한 지도 시각화 (o)
 7. 범죄율에 대한 지도 시각화 (o)
 8. 경찰서별 검거율과 구별 범죄율 시각화 (o)
 9. 소감 (o)

이름 : 구명회
학번 : 20154215
학과 : 컴퓨터 공학과



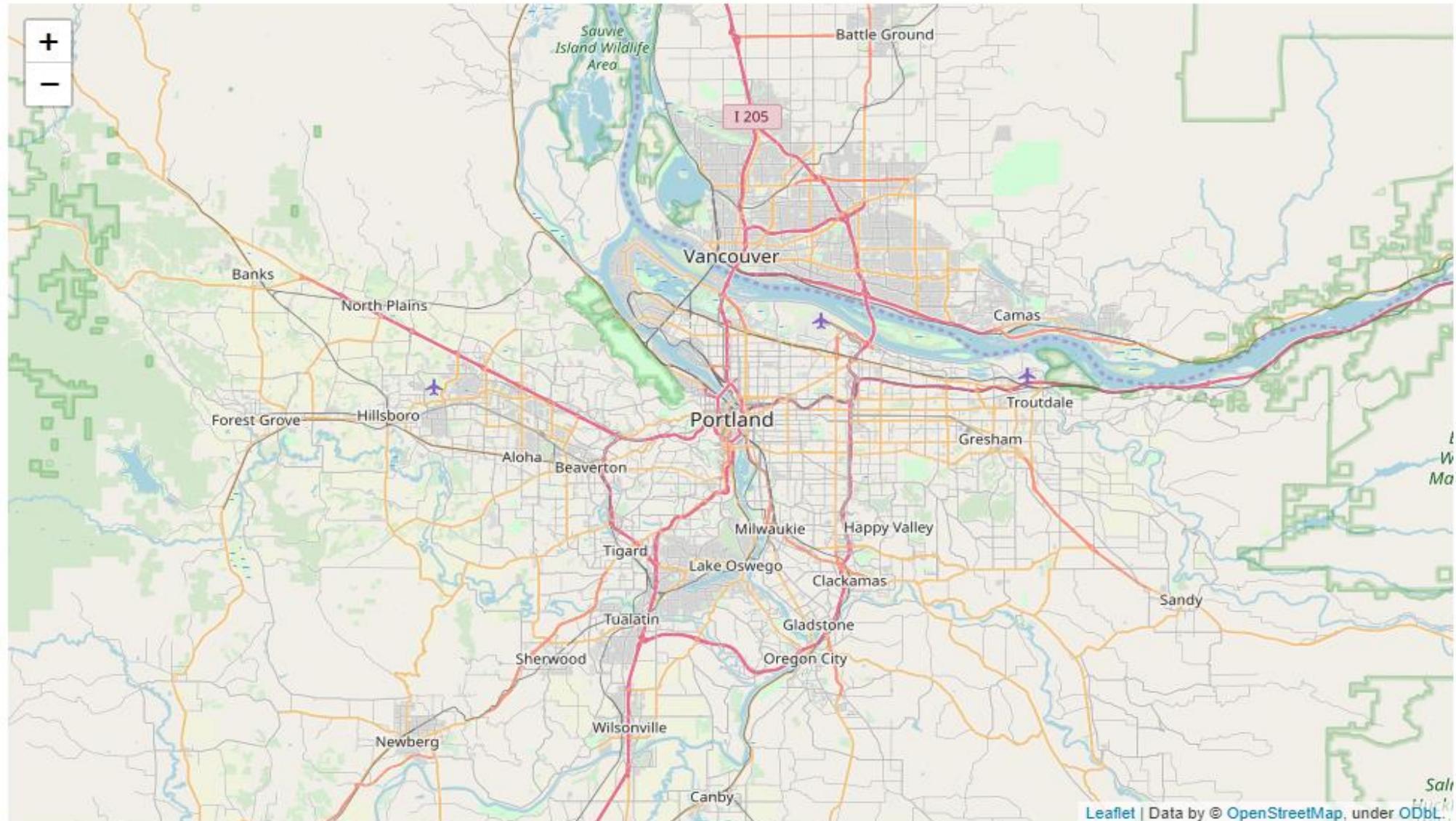


6. Folium을 이용한 지도 시각화

```
In [38]: # pip install folium ..... folium 라이브러리 설치  
import folium
```

```
In [39]: # 위도와 경도 정보를 가지고 지도를 그렸습니다.  
# IE는 지도 표현이 원활하지 않아 CHROME으로 사용했습니다.  
map_osm = folium.Map(location=[45.5236, -122.6750])  
map_osm
```

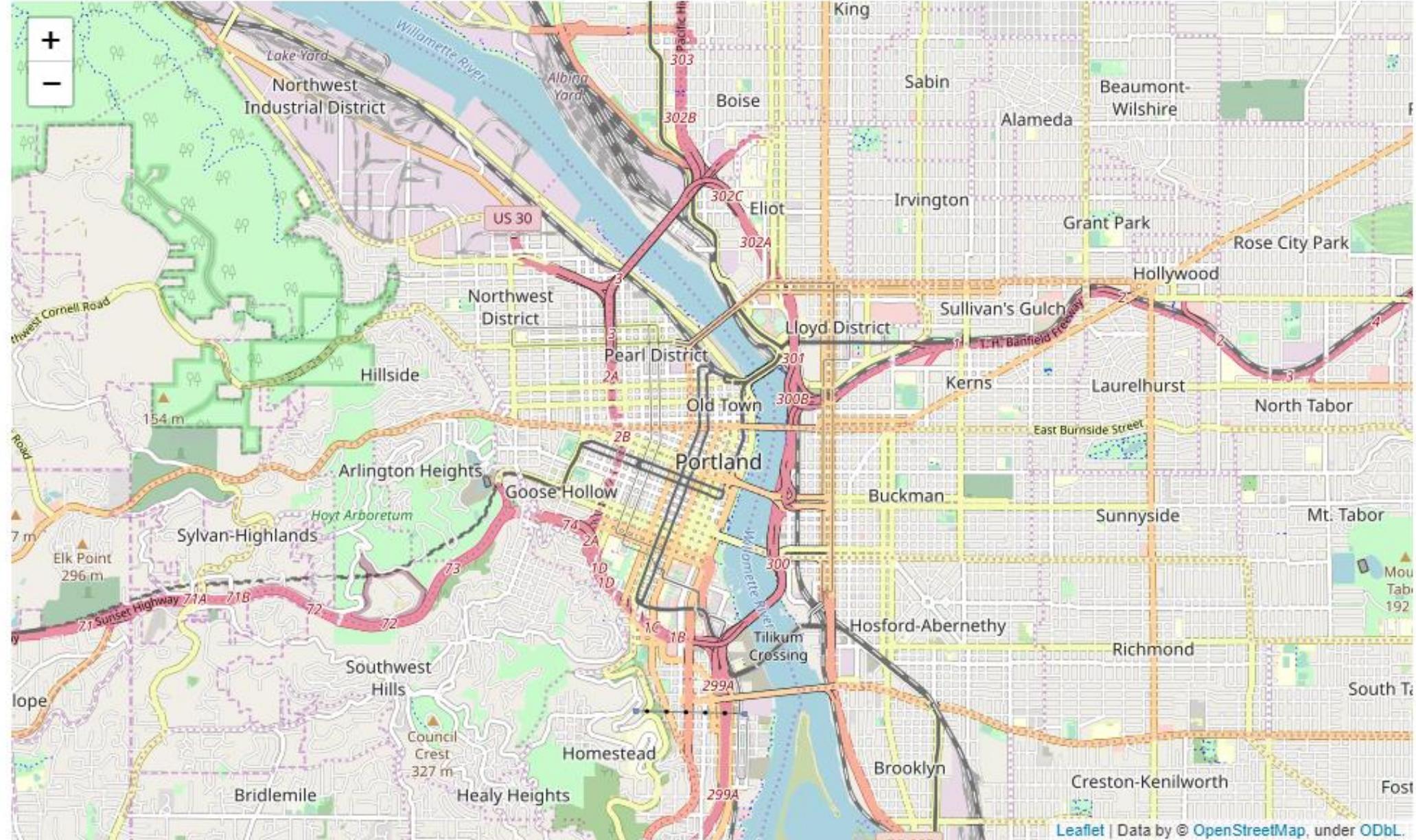
Out [39]:



In [40]: # zoom_start 옵션으로 확대 비율을 설정했습니다.

```
stamen = folium.Map(location=[45.5236, -122.6750], zoom_start=13)  
stamen
```

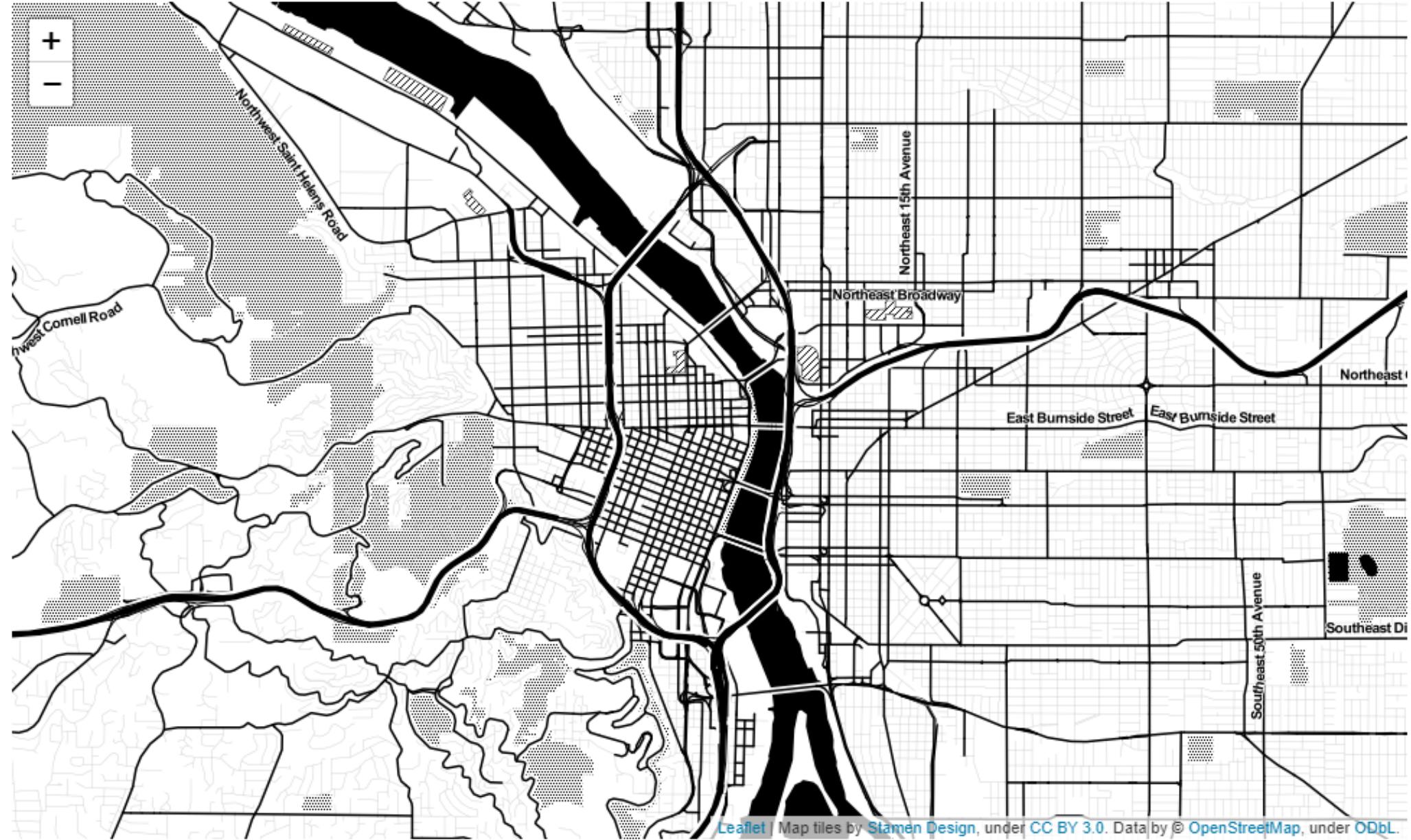
Out [40] :



In [41]: # zoom_start와 tiles 옵션을 사용해 확대 비율과 지도의 모양을 변경합니다

```
stamen = folium.Map(location=[45.5236, -122.6750], tiles='Stamen Toner', zoom_start=13)  
stamen
```

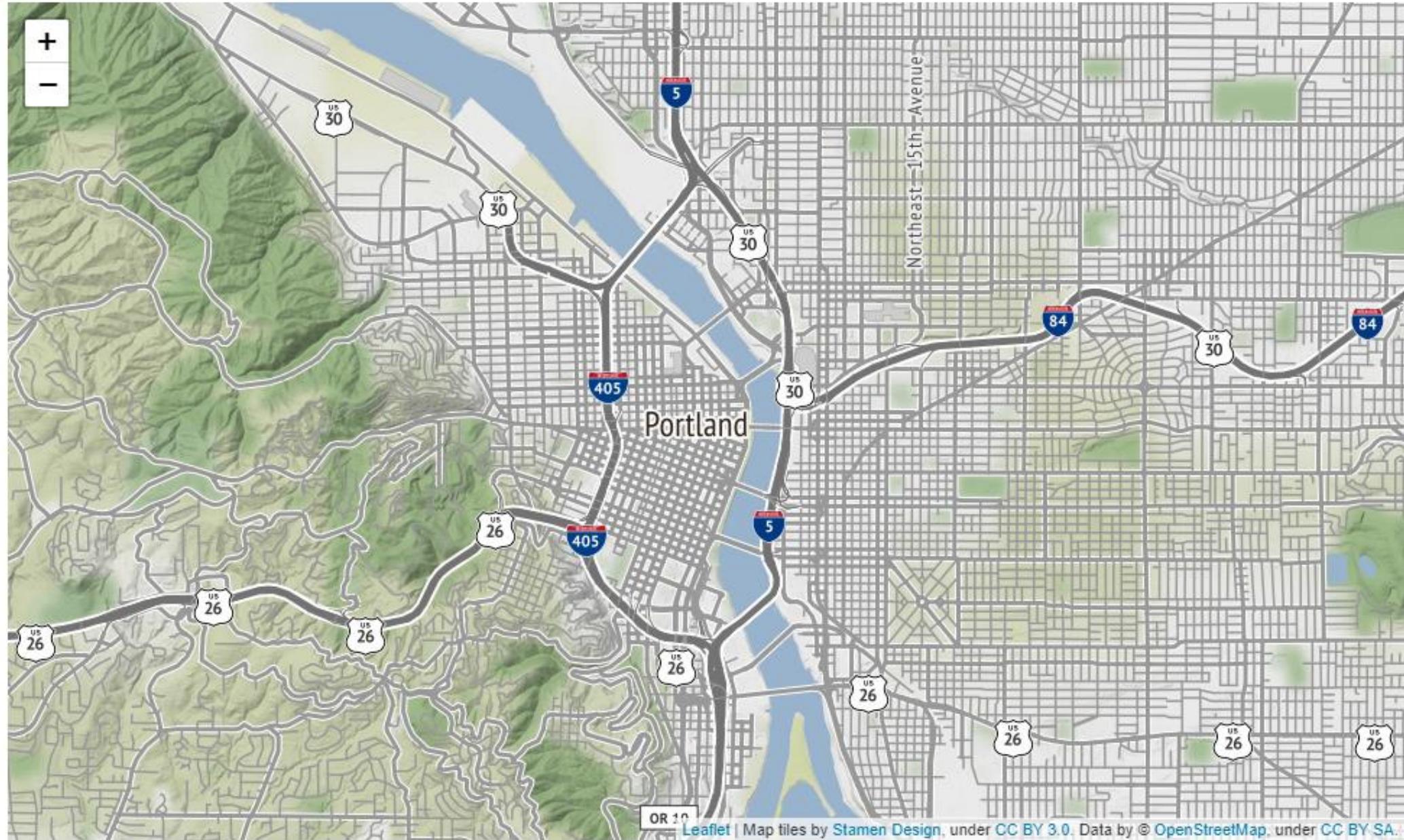
Out [41]:



In [42]: # zoom_start와 tiles 옵션을 사용해 확대 비율과 지도의 모양을 변경합니다

```
stamen = folium.Map(location=[45.5236, -122.6750], tiles='Stamen Terrain', zoom_start=13)  
stamen
```

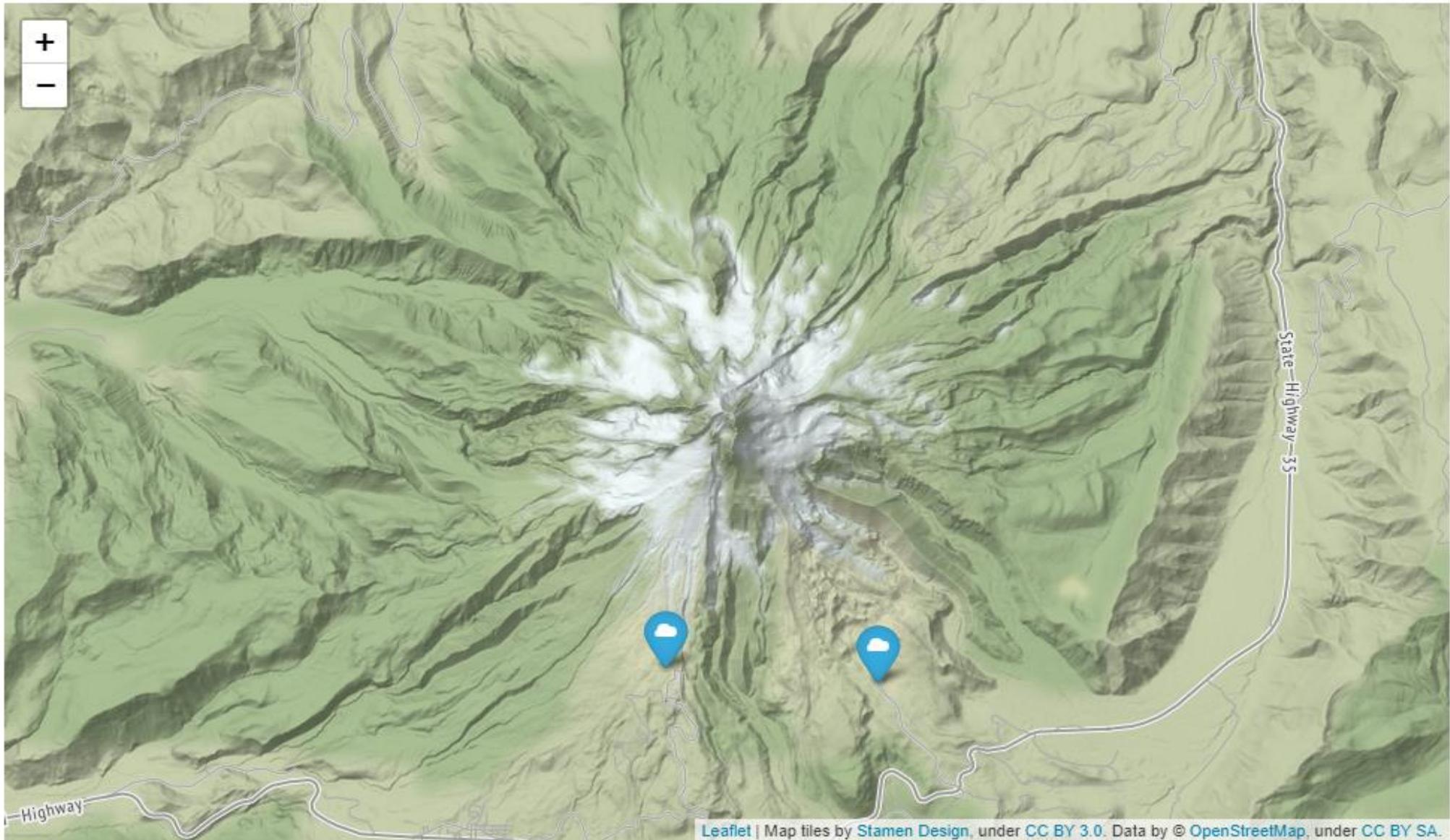
Out [42]:



In [43]:

```
# zoom_start와 tiles 옵션을 사용해 확대 비율과 지도의 모양을 변경합니다  
# Marker 옵션을 사용해 지정한 위치에 마크를 표시했습니다.  
map_1 = folium.Map(location=[45.372, -121.6972], zoom_start=12, tiles='Stamen Terrain')  
folium.Marker([45.3288, -121.6625], popup='Mt. Hood Meadows', icon=folium.Icon(icon='cloud')).add_to(map_1)  
folium.Marker([45.3311, -121.7113], popup='Timberline Lodge', icon=folium.Icon(icon='cloud')).add_to(map_1)  
map_1
```

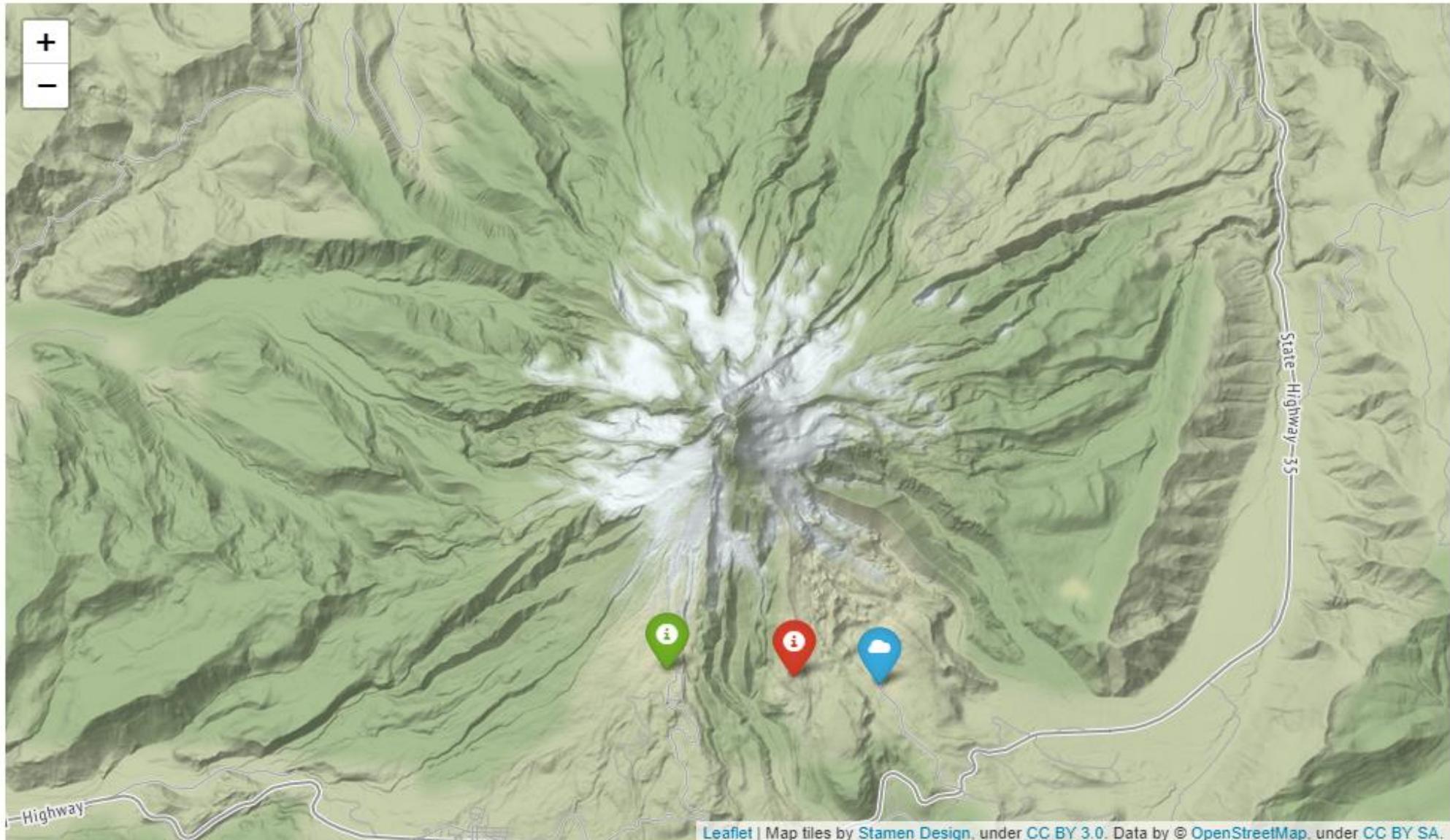
Out [43]:



In [44]:

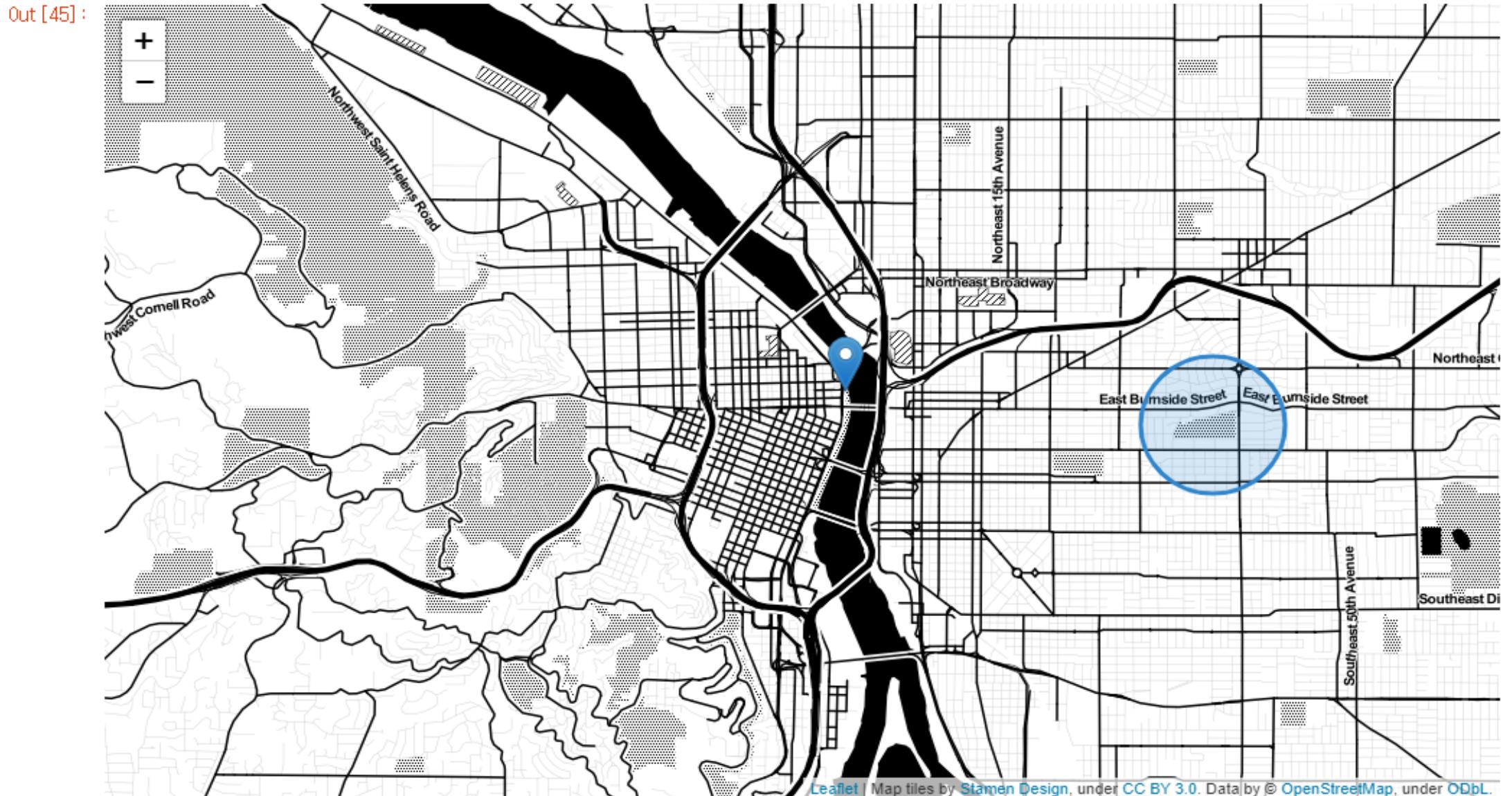
```
# zoom_start와 tiles 옵션을 사용해 확대 비율과 지도의 모양을 변경합니다  
# Marker 옵션을 사용해 지정한 위치에 마크를 표시했습니다.  
map_1 = folium.Map(location=[45.372, -121.6972], zoom_start=12, tiles='Stamen Terrain')  
folium.Marker([45.3288, -121.6625], popup='Mt. Hood Meadows', icon=folium.Icon(icon='cloud')).add_to(map_1)  
folium.Marker([45.3311, -121.7113], popup='Timberline Lodge', icon=folium.Icon(color='green')).add_to(map_1)  
folium.Marker([45.3300, -121.6823], popup='Some Other Location', icon=folium.Icon(color='red',icon='info-sign')).add_to(map_1)  
map_1
```

Out [44]:



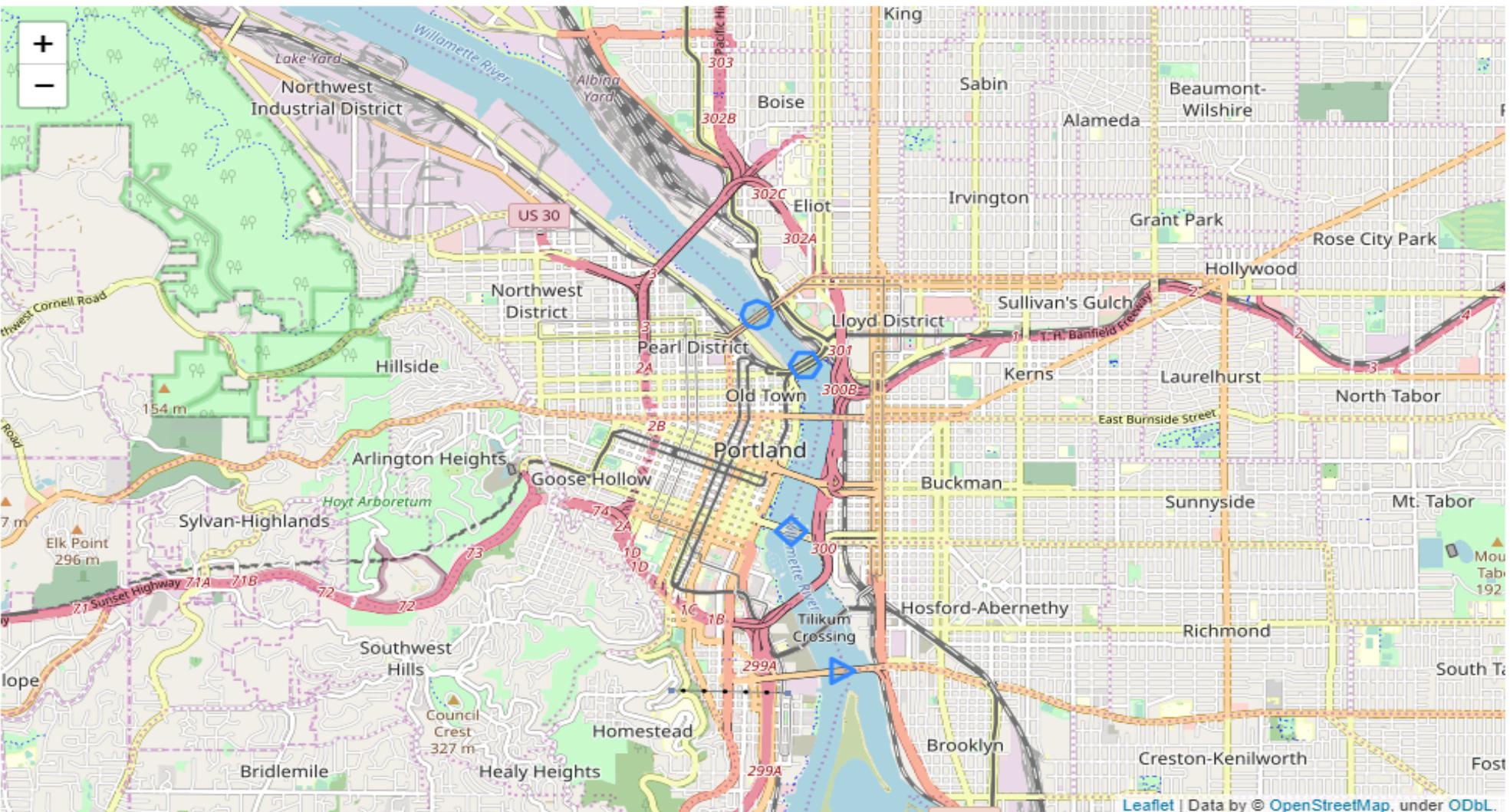
In [45]:

```
# zoom_start와 tiles 옵션을 사용해 확대 비율과 지도의 모양을 변경합니다
# Marker 옵션을 사용해 지정한 위치에 마크를 표시했습니다.
# CircleMarker 옵션을 사용해 지정한 반경과 색상의 원을 그렸습니다.
map_2 = folium.Map(location=[45.5236, -122.6750], tiles='Stamen Toner', zoom_start=13)
folium.Marker([45.5244, -122.6699], popup='The Waterfront').add_to(map_2)
folium.CircleMarker([45.5215, -122.6261], radius=50, popup='Laurelhurst Park', color='#3186cc', fill_color='#3186cc').add_to(map_2)
map_2
```



```
In [46]: # RegularPolygonMarker 옵션을 사용해 지도를 출렁였습니다.  
map_5 = folium.Map(location=[45.5236, -122.6750], zoom_start=13)  
folium.RegularPolygonMarker([45.5012, -122.6655], popup='Ross Island Bridge', fill_color='#132b5e',  
                           number_of_sides=3, radius=10).add_to(map_5)  
folium.RegularPolygonMarker([45.5132, -122.6708], popup='Hawthorne Bridge', fill_color='#45647d',  
                           number_of_sides=4, radius=10).add_to(map_5)  
folium.RegularPolygonMarker([45.5275, -122.6692], popup='Steel Bridge', fill_color='#769d96',  
                           number_of_sides=6, radius=10).add_to(map_5)  
folium.RegularPolygonMarker([45.5318, -122.6745], popup='Broadway Bridge', fill_color='#769d96',  
                           number_of_sides=8, radius=10).add_to(map_5)  
map_5
```

Out [46] :



```
In [47]: # pip install folium >>> folium 라이브러리 설치  
# folium은 시각화 도구  
# folium과 pandas를 임포트  
import folium  
import pandas as pd
```

```
In [48]: # data를 pandas로 읽어서 state_unemployment 변수에 저장합니다.  
# .head() : 인자값만큼 행을 출력한다. 인자를 default로 주면 pandas data의 첫 5행만 출력  
state_unemployment = '../data/02_folium_US_Unemployment_Oct2012.csv'  
state_data = pd.read_csv(state_unemployment)  
state_data.head()
```

Out [48] :

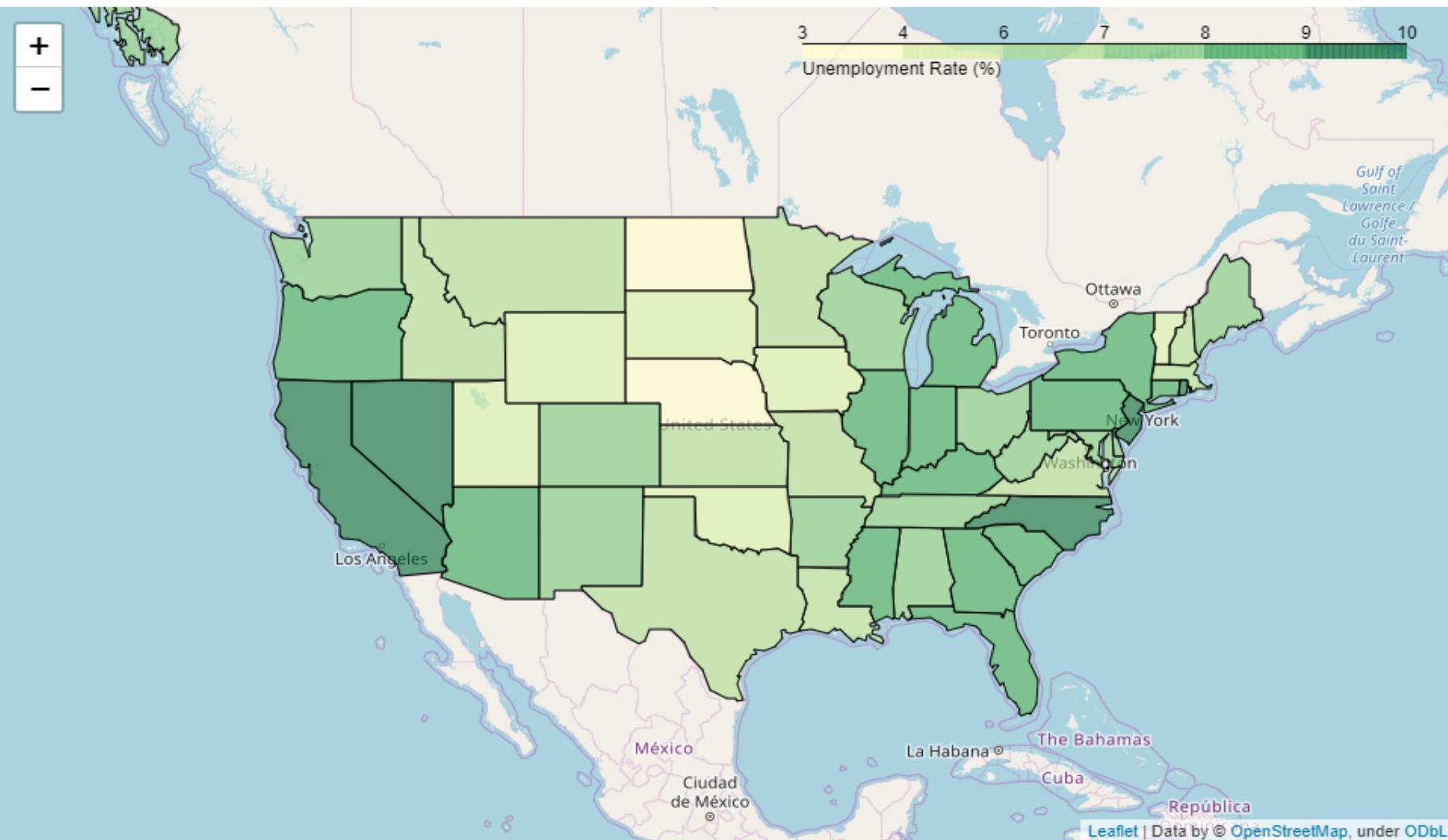
	State	Unemployment
0	AL	7.1
1	AK	6.8
2	AZ	8.1
3	AR	7.2
4	CA	10.1

In [49]: # json파일안에는 여러 좌표들, 특히 위도와 경도 좌표들이 입력되어 있습니다.
위도와 경도를 따라 주를 구별하기 위한 경계선을 그렸습니다.
state_geo = '../data/02. folium_us-states.json'

```
map = folium.Map(location=[40, -98], zoom_start=4)
map.choropleth(geo_data=state_geo, data=state_data, columns=['State', 'Unemployment'],
               key_on='feature.id', fill_color='YlGn', legend_name='Unemployment Rate (%)')
map
```

C:\Users\herin\Anaconda3\lib\site-packages\folium\folium.py:415: FutureWarning: The choropleth method has been deprecated. Instead use the new Choropleth class, which has the same arguments. See the example notebook 'GeoJSON_and_choropleth' for how to do this.
FutureWarning

Out [49]:

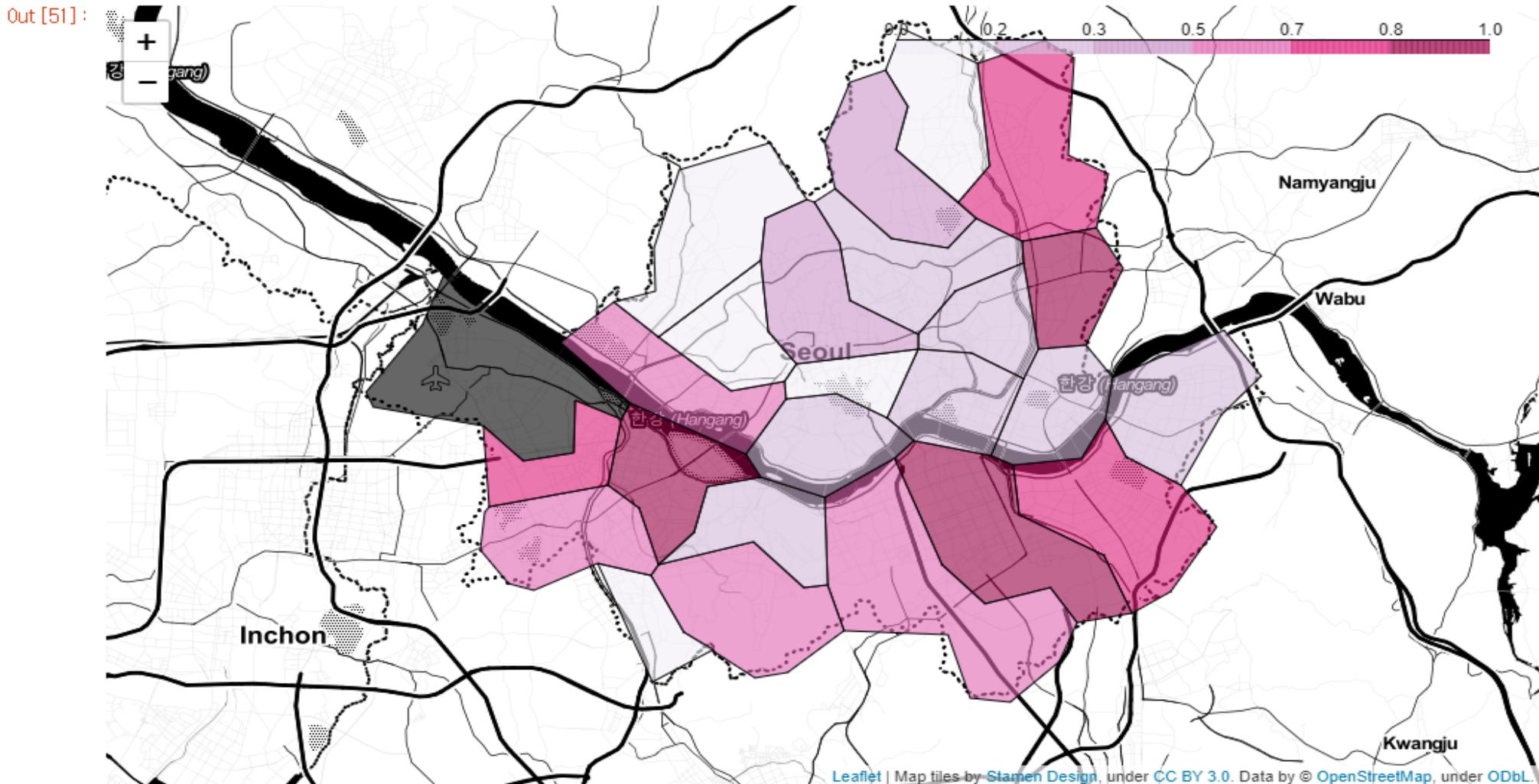


7. 범죄율에 대한 지도 시각화

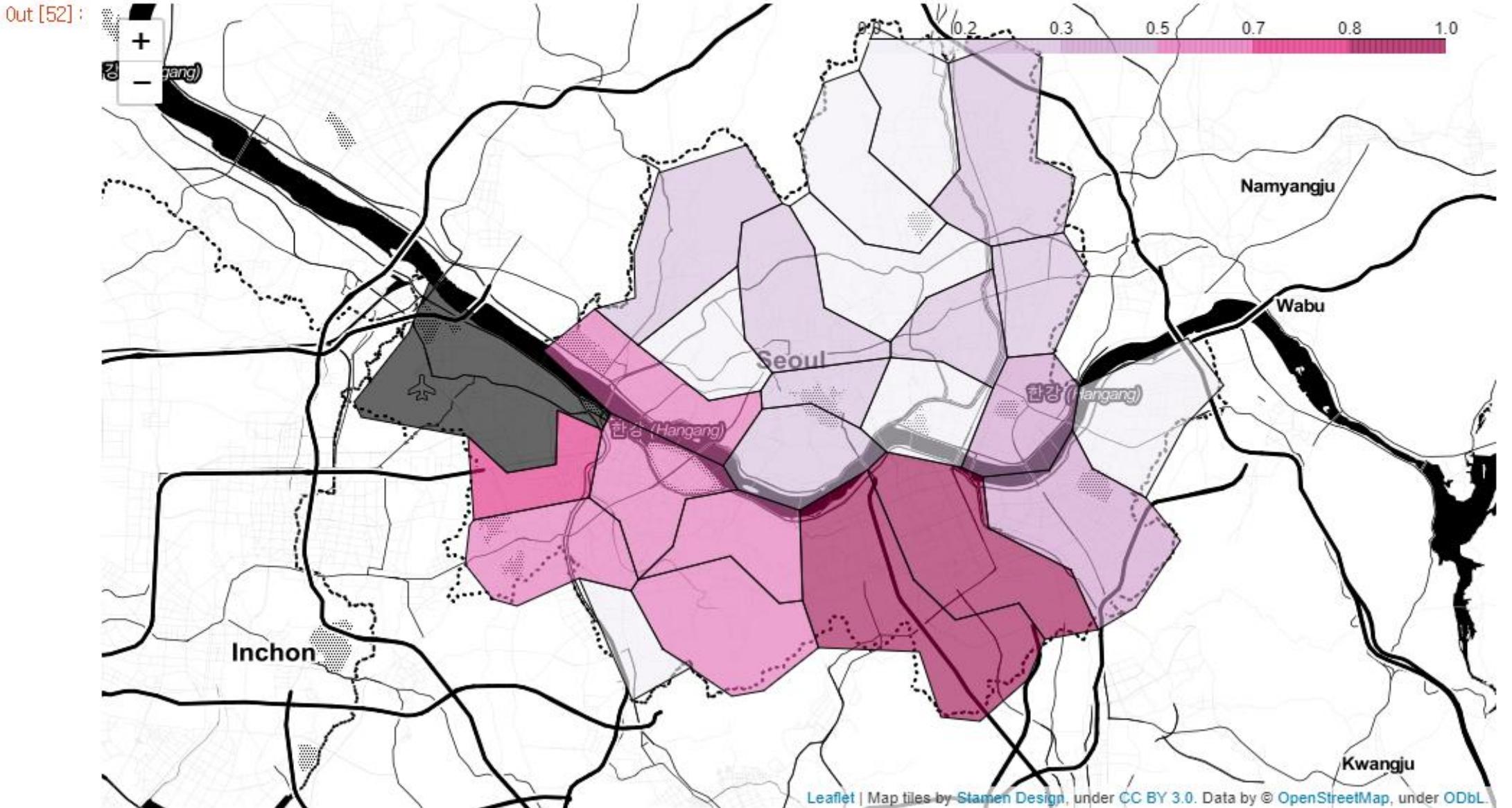


```
In [50]: # .load : json 파일을 읽어옵니다.  
# encoding = 'utf-8' : 인코딩 data를 utf-8을 사용하겠다  
import json  
geo_path = '../data/02. skorea_municipalities_geo_simple.json'  
geo_str = json.load(open(geo_path, encoding='utf-8'))
```

```
In [51]: # 서울시 살인건수를 알아보기 위해 서울시의 중심의 위도와 경도를 입력하고 경계선을 그렸습니다.  
map = folium.Map(location=[37.5502, 126.982], zoom_start=11, tiles='Stamen Toner')  
  
map.choropleth(geo_data = geo_str, data = crime_anal_norm['살인'],  
                columns = [crime_anal_norm.index, crime_anal_norm['살인']], fill_color = 'PuRd', key_on = 'feature.id')  
map
```



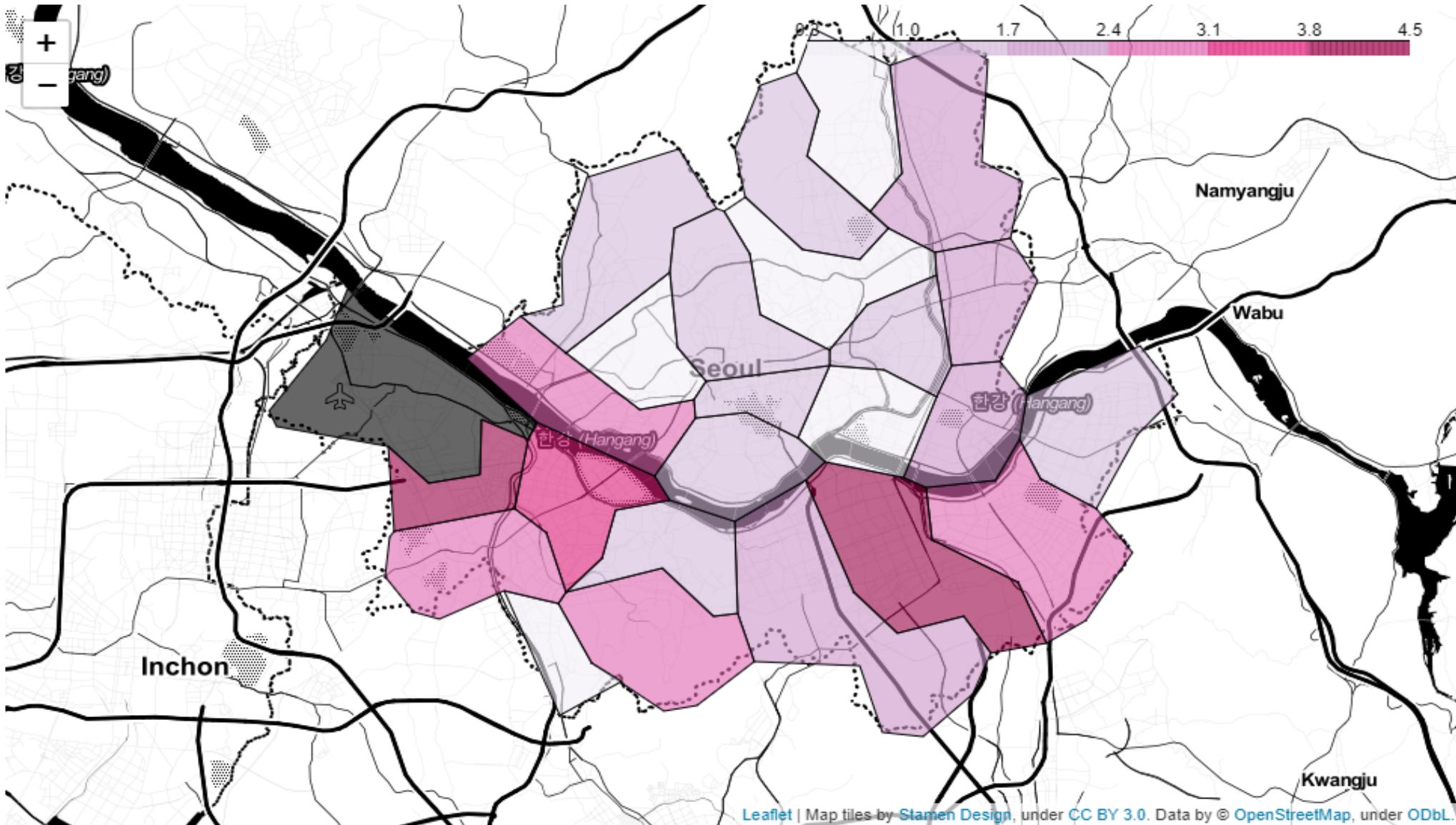
```
In [52]: # 서울시 강간건수를 알아보기 위해 서울시의 중심의 위도와 경도를 입력하고 경계선을 그렸습니다.  
map = folium.Map(location=[37.5502, 126.982], zoom_start=11, tiles='Stamen Toner')  
  
map.choropleth(geo_data = geo_str, data = crime_anal_norm['강간'], columns = [crime_anal_norm.index, crime_anal_norm['강간']],  
                fill_color = 'PuRd', key_on = 'feature.id')  
map
```



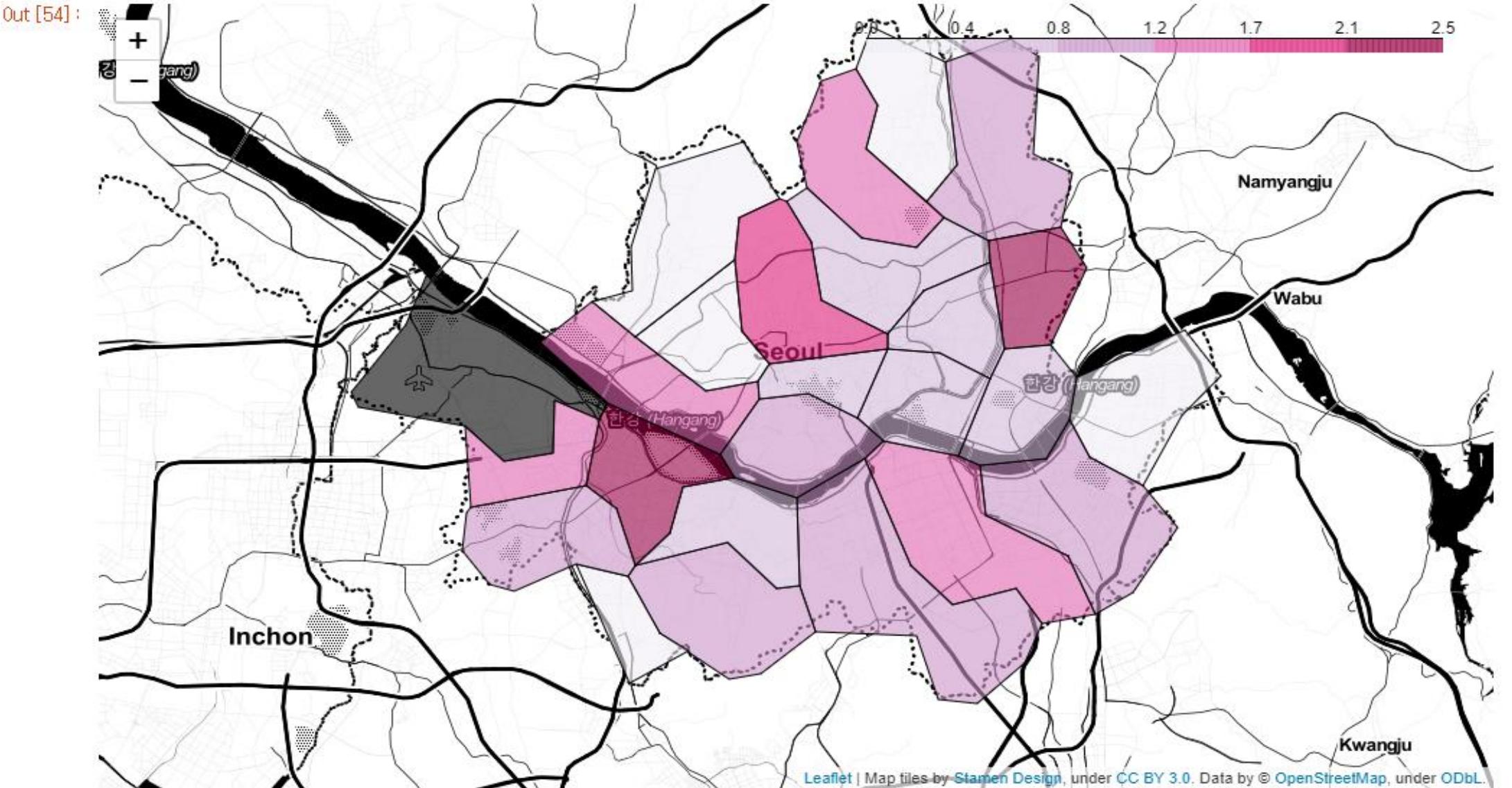
In [53]:

```
# 범죄 발생 건수 전체를 지도상에 출력합니다.  
# 위의 살인과 강간 발생 건수가 높았던 지역이 여전히 높게 출력됩니다.  
map = folium.Map(location=[37.5502, 126.982], zoom_start=11, tiles='Stamen Toner')  
  
map.choropleth(geo_data = geo_str, data = crime_anal_norm['범죄'], columns = [crime_anal_norm.index, crime_anal_norm['범죄']],  
                fill_color = 'PuRd', key_on = 'feature.id')  
map
```

Out [53]:



```
In [54]: # 인구수를 고려해서 적절한 값을 계산해 살인 발생 건수를 다시 지도상에 출력했습니다.  
tmp_crriminal = crime_anal_norm['살인'] / crime_anal_norm['인구수'] * 1000000  
  
map = folium.Map(location=[37.5502, 126.982], zoom_start=11, tiles='Stamen Toner')  
  
map.choropleth(geo_data = geo_str, data = tmp_crriminal, columns = [crime_anal.index, tmp_crriminal],  
                fill_color = 'PuRd', key_on = 'feature.id')  
map
```



```
In [55]: # 인구수를 고려해서 적절한 값을 계산해 전체 범죄 발생 건수를 다시 지도상에 출력했습니다.
```

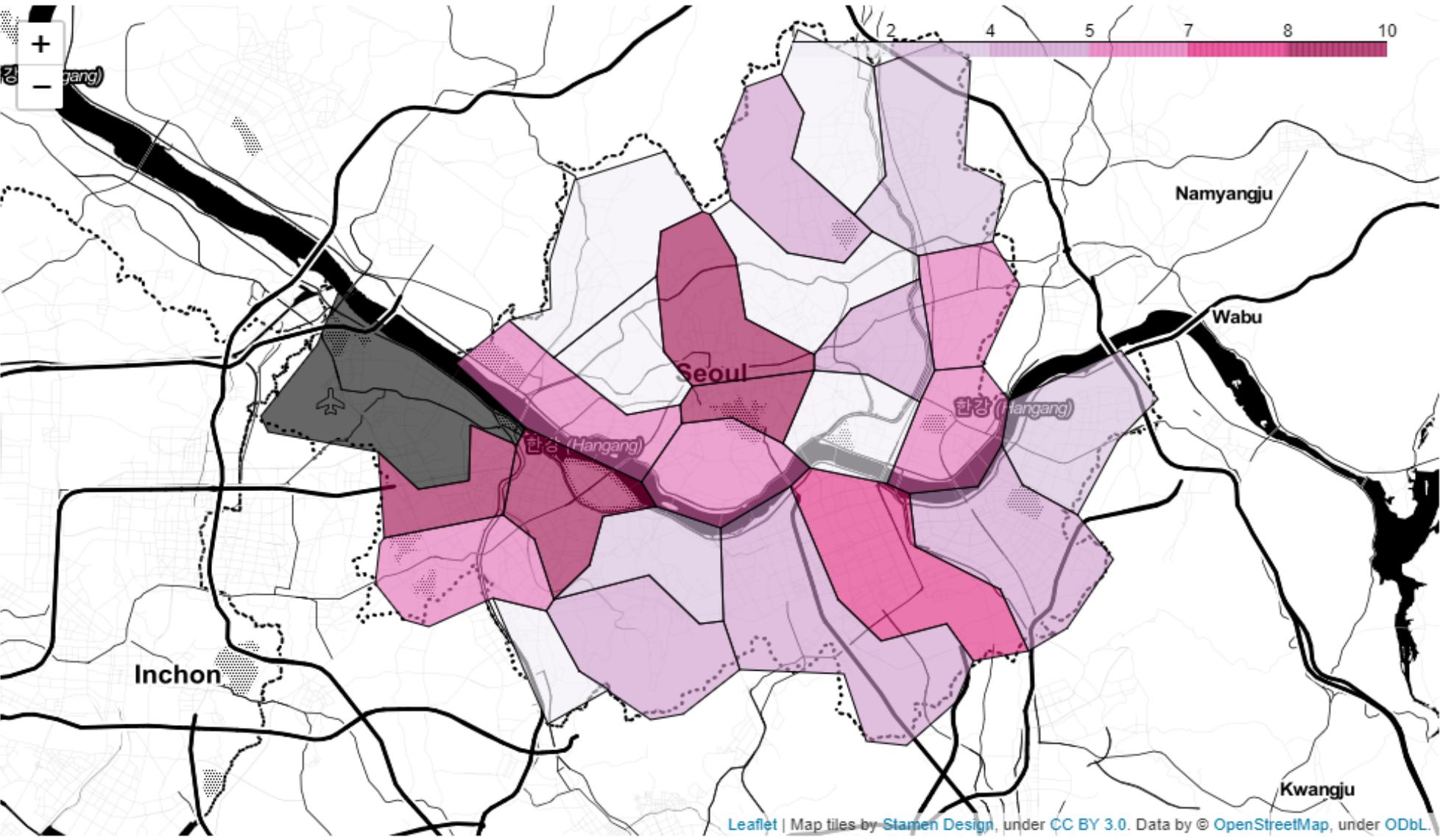
```
tmp_crriminal = crime_anal_norm['범죄'] / crime_anal_norm['인구수'] * 1000000
```

```
map = folium.Map(location=[37.5502, 126.982], zoom_start=11, tiles='Stamen Toner')
```

```
map.choropleth(geo_data = geo_str, data = tmp_crriminal, columns = [crime_anal.index, tmp_crriminal],  
               fill_color = 'PuRd', key_on = 'feature.id')
```

```
map
```

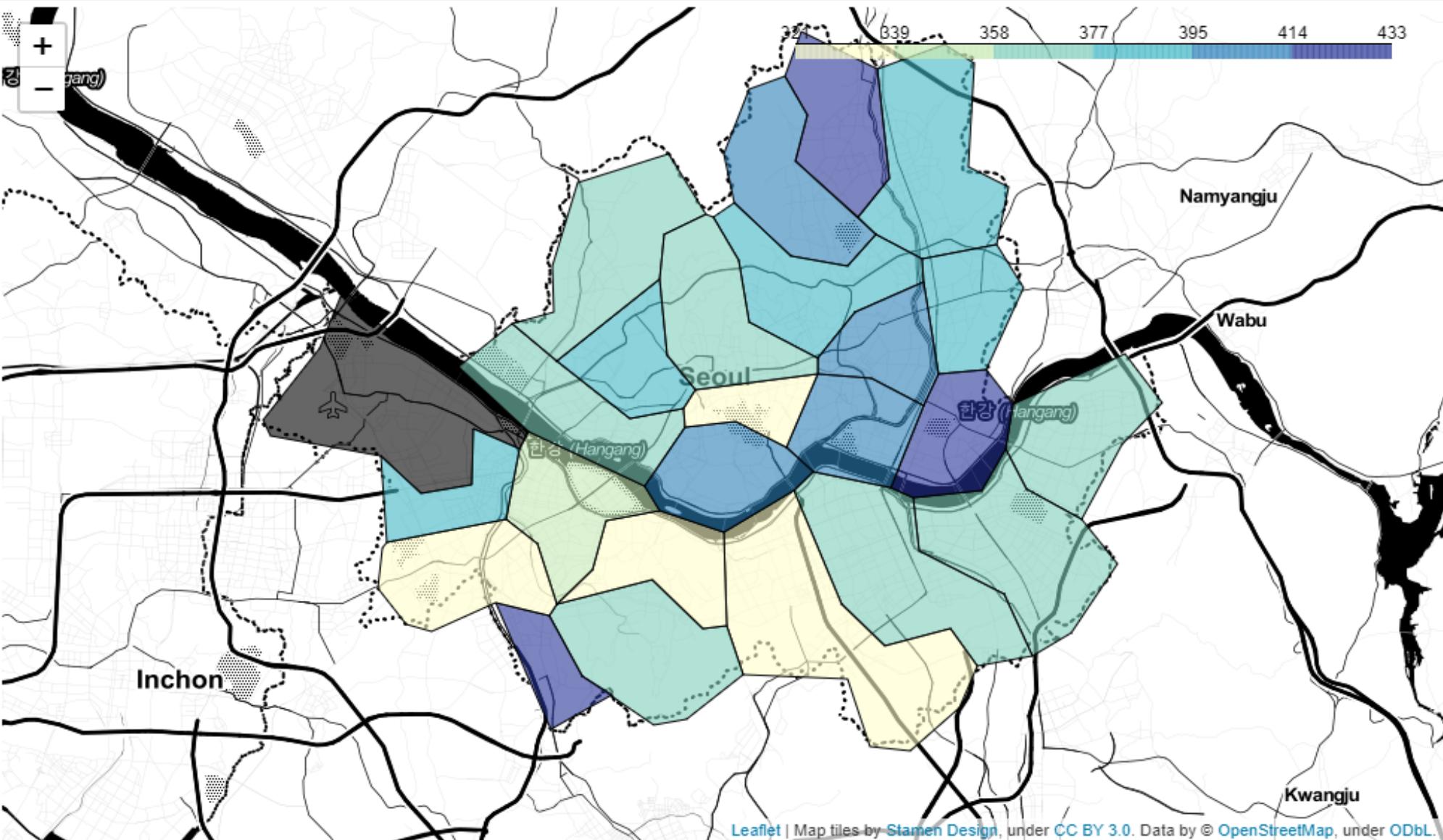
```
Out [55]:
```



```
In [56]: # 서울시 경계경수를 알아보기 위해 서울시의 중심의 위도와 경도를 입력하고 경계선을 그렸습니다.  
map = folium.Map(location=[37.5502, 126.982], zoom_start=11, tiles='Stamen Toner')
```

```
map.choropleth(geo_data = geo_str,  
                data = crime_anal_norm['경계'], columns = [crime_anal_norm.index, crime_anal_norm['경계']],  
                fill_color = 'YlGnBu', key_on = 'feature.id')  
map
```

Out [56]:



8. 경찰서별 검거율과 구별 범죄율 시각화



```
In [57]: # 현재 범죄 발생목록 중에서 검거만 출력합니다.
```

```
crime_anal_raw['lat'] = station_lat
```

```
crime_anal_raw['lng'] = station_lng
```

```
col = ['살인 검거', '강도 검거', '강간 검거', '절도 검거', '폭력 검거']
```

```
tmp = crime_anal_raw[col] / crime_anal_raw[col].max()
```

```
crime_anal_raw['검거'] = np.sum(tmp, axis=1)
```

```
crime_anal_raw.head()
```

```
Out[57]:
```

	관서명	살인 발생	살인 검거	강도 발생	강도 검거	강간 발생	강간 검거	절도 발생	절도 검거	폭력 발생	폭력 검거	구별	lat	lng	검거
0	중부서	2	2	3	2	105	65	1395	477	1355	1170	증구	37.563646	126.989580	1.275416
1	종로서	3	3	6	5	115	98	1070	413	1278	1070	종로구	37.575558	126.984867	1.523847
2	남대문서	1	0	6	4	65	46	1153	382	869	794	증구	37.554758	126.973498	0.907372
3	서대문서	2	2	5	4	154	124	1812	738	2056	1711	서대문구	37.564785	126.966776	1.978299
4	혜화서	3	2	5	4	96	63	1114	424	1015	861	종로구	37.571853	126.998914	1.198382

```
In [58]: # 각 경찰서의 위도와 경도를 이용해서 Marker로 경찰서의 위치를 확인했습니다.
```

```
map = folium.Map(location=[37.5502, 126.982], zoom_start=11)
```

```
for n in crime_anal_raw.index:
```

```
    folium.Marker([crime_anal_raw['lat'][n], crime_anal_raw['lng'][n]]).add_to(map)
```

```
map
```

```
Out [58] :
```

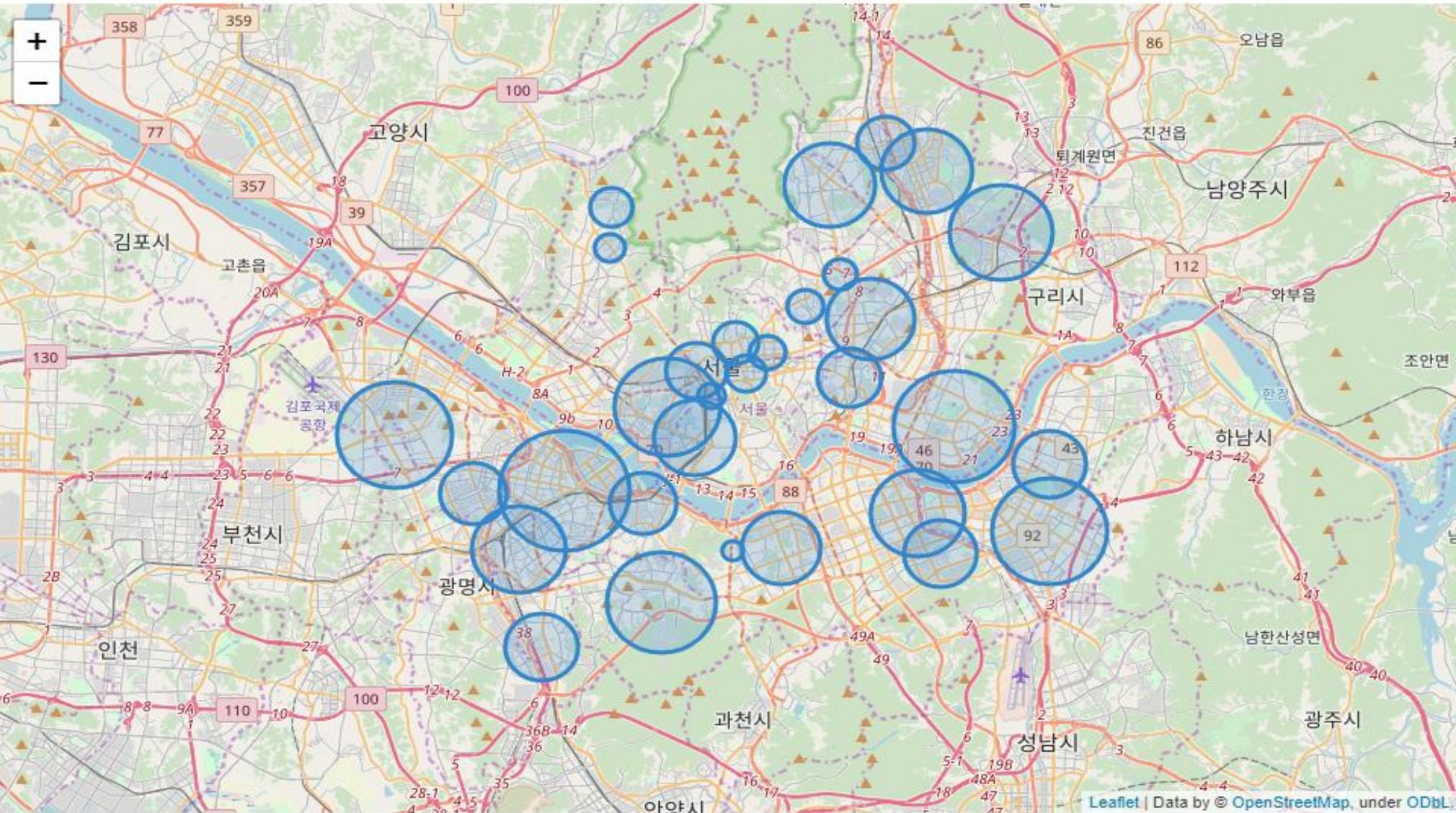


```
In [59]: # 각 경찰서의 경거율을 원의 넓이로 표현했습니다.  
map = folium.Map(location=[37.5502, 126.982], zoom_start=11)
```

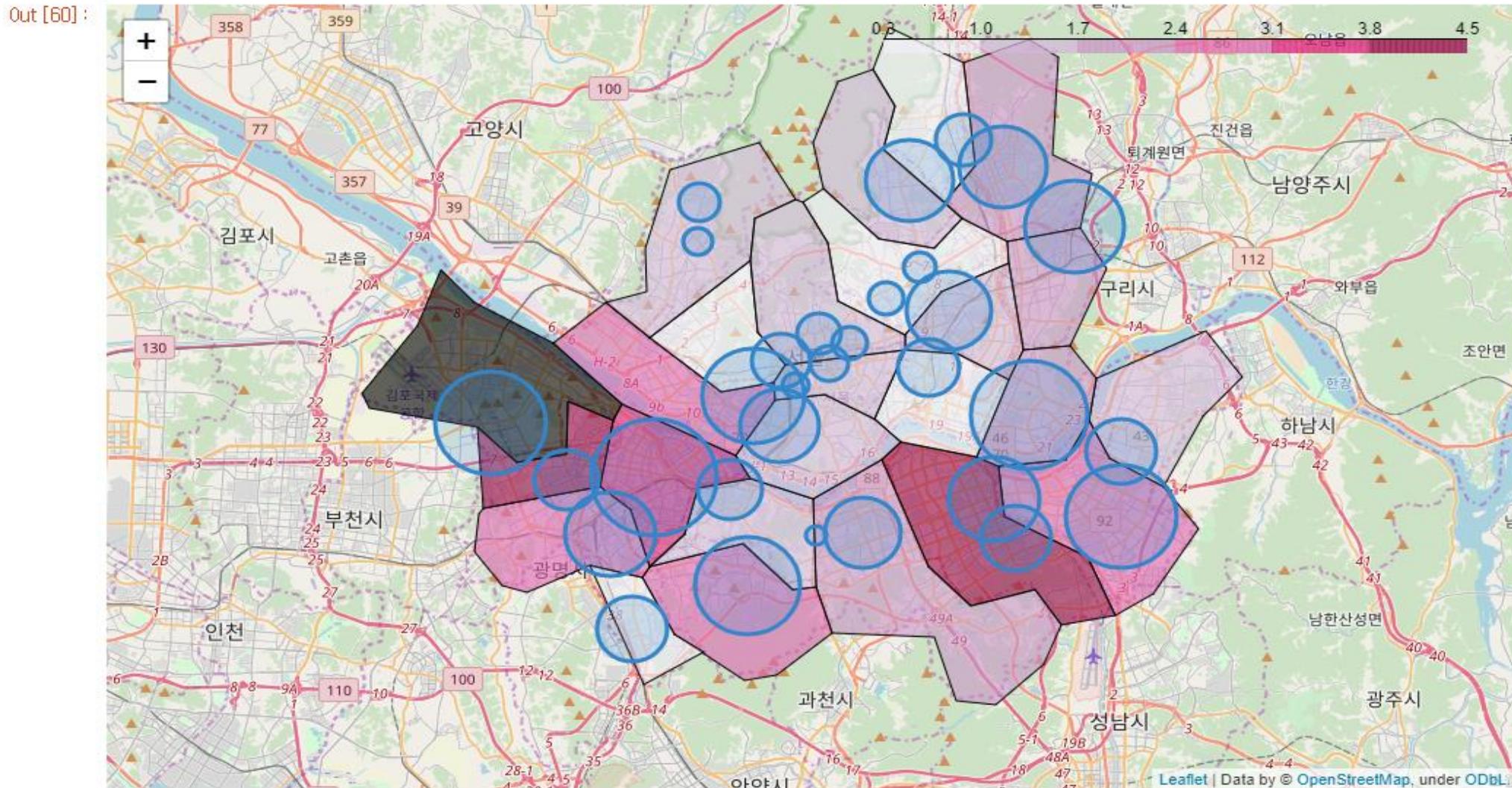
```
for n in crime_anal_raw.index:  
    folium.CircleMarker([crime_anal_raw['lat'][n], crime_anal_raw['lng'][n]], radius = crime_anal_raw['경거'][n]*10,  
                        color='#3186cc', fill_color='#3186cc', fill=True).add_to(map)
```

```
map
```

```
Out [59]:
```



```
In [60]: # 경찰서의 경거를 위에 범죄 발생 건수를 블루색으로 표현했습니다.  
map = folium.Map(location=[37.5502, 126.982], zoom_start=11)  
  
map.choropleth(geo_data = geo_str, data = crime_anal_norm['범죄'], columns = [crime_anal_norm.index, crime_anal_norm['범죄']],  
                fill_color = 'PuRd', key_on = 'feature.id')  
  
for n in crime_anal_raw.index:  
    folium.CircleMarker([crime_anal_raw['lat'][n], crime_anal_raw['lng'][n]],  
                        radius = crime_anal_raw['경거'][n]*10, color='#3186cc', fill_color='#3186cc', fill=True).add_to(map)  
  
map
```



9. 소감

2장 뒤쪽 지도에 표현하는 과제입니다. 전에 제출했던 부분은 생략했습니다.
사실 저번주에 냈던 2장 ppt와 합쳐서 제출하려고 했는데 생각해보니까 코드에
주석을 다 달아놓고 변수 통일을 안해서 그냥 따로 제작했습니다.

2장을 마치며, 느낀점은 저번에 다른 사람이 말했던 소감이긴 하지만 확실히 데이터 출력 형태가
그래프나 지도와 같은 형태라면 눈으로 그 차이를 확실히 알 수 있어서 좋은것 같습니다.
저번학기에 시각화가 중요하다고 꼭 해보라하셨던 정○○ 교수님의 말씀이 떠올랐습니다.

이제 folium이나 구글 맵 api를 사용해서 지도를 표현하는 방법을 알았습니다.
나중에 기회가 된다면 우리 지역에 대한 통계도 따로 표현해보고 싶습니다.

